# Fișa 06 C# – clase: noțiuni de bază

1. **Mod de definire**: clasele sunt structuri de date asemănătoare celor construite folosind **struct**. Spre deosebire de structuri (care conțin doar date) o clasă poate conține și funcții și permite operații mai complexe decât cele posibile cu o structură (moștenire și altele). Pentru a defini o clasă se folosește o construcție de forma:

//clasă definită de utilizator

class nume\_clasa

{

//definiții variabile (numite și "câmpuri" sau "variabile membru")

[modificator acces] tip nume\_variabilă;

//definiții funcții (numite și "metode" sau "funcții membru")

[modificator acces] tip nume\_funcție( [listă\_parametri] ) { ... }

}

1. **Observații**:
   1. Variabilele se declară în mod obișnuit existând posibilitatea de a folosi pentru fiecare un **modificator de acces**. Acesta precizează locul din program în care variabila sau funcția este vizibilă, în care poate fi folosită. Modificatorii de acces disponibili și semnificațiile lor sunt:
      * ***public***: acces nelimitat;
      * ***private***: acces limitat la propria clasă;
      * ***protected***: acces limitat la propria clasă și la clasele derivate (clasele „fiu”);
      * ***internal***: acces limitat la programul în care este definită clasa;
      * ***protected internal***: acces limitat la programul în care e definită clasa și la clasele derivate;
      * dacă modificatorul de acces lipsește, se consideră că este ***private***.
   2. Variabilele definite într-o clasă se numesc **câmpuri** sau **variabile membru** (ale clasei).
   3. Funcțiile definite într-o clasă se numesc **metode** sau **funcții membru**.
   4. După definirea clasei se pot declara variabile de tip clasă. Acestea se numesc **obiecte** sau **instanțe** ale clasei. Putem spune că definiția clasei este un model, un „proiect”, după care se construiesc în memorie aceste obiecte. Operația prin care se construiește un obiect (se alocă memorie pentru el pe baza unei clase) se numește **instanțiere** și se face folosind operatorul ***new***.
   5. Având o clasă definită și un obiect construit pe baza acesteia, accesul la datele sau funcțiile obiectului se face cu construcția: **nume\_obiect.nume\_câmp** sau **nume\_obiect.nume\_metodă**.
   6. Într-un program C# se pot defini oricâte clase. Dacă sunt mai multe atunci una singură dintre ele va conține metoda **Main** și aceasta se va executa prima (este „punctul de intrare” în program sau de pornire a programului).
2. **Exemplu**:

using System;

class Copil

{

//declarare campuri

public string nume;

int varsta;

string sex;

//declarare metode

public Copil(string nume = "necunoscut", int varsta = 0, string sex = "necunoscut")

{

this.nume = nume;

this.varsta = varsta;

this.sex = sex;

}

void plange() { Console.WriteLine("\n" + nume + " plange!"); }

public void afisare() { Console.WriteLine("\nNume: "+nume+", varsta: "+varsta+", sex: "+sex); }

}

class Program

{

static void Main()

{

Copil c0;

Copil c1 = new Copil();

Copil c2 = new Copil("Vasile");

Copil c3 = new Copil("Maria", 12);

Copil c4 = new Copil("Mia", 18, "f");

Copil c5 = new Copil(varsta:22, sex:"f");

Copil c6 = new Copil("Petre", sex: "m");

//c0.nume = "Ion"; //Eroare! -> c0 nu este instanțiat!

c1.nume = "Ion";

//c1.varsta = 29; //Eroare! -> câmpul "varsta" este privat!

//c0.plange(); //Eroare! -> metoda "plange" este privată!

//c0.afisare(); //Eroare! -> obiectul c0 este neinstanțiat!

c1.afisare();

c2.afisare();

c3.afisare();

c4.afisare();

c5.afisare();

c6.afisare();

//gata

Console.WriteLine("\n\n\n");

}

}